SCREEN PRINT COMMAND

A useful ProDOS command that dumps a 40- or 80-column text screen to a printer.

Printer Dump is a ProDOS utility that enables you to dump the contents of a text screen to a printer. The program installs itself as an external command so you can run it from BASIC, from the monitor, or from a program. It automatically switches between 40- and 80-column modes.

USING THE PROGRAM

To install the DUMP utility, type

BRUN PRINTER.DUMP

Once you've done that, you have a new ProDOS command. From BASIC and the Monitor, type DUMP to dump the screen to your printer. From inside a BASIC program, use

PRINT CHR$(4);"DUMP"

The CHR$(4) tells Applesoft that you are issuing a ProDOS command; you issue most ProDOS commands (CATALOG, BSAVE, etc.) from your BASIC programs in this manner.

```
10 HOME
20 GET A$
30 IF A$ = CHR$(5) THEN END
40 IF A$ = CHR$(4) THEN PRINT CHR$(4)"DUMP" : GOTO 20
50 PRINT A$; : GOTO 20
```

EXAMPLE 1: Using the Screen Dump Command

Example 1 uses the DUMP command. It lets you type on the screen and then print. You can move the cursor using the Escape mode, quit by pressing Control-E, and print the screen by pressing Control-D.

ENTERING THE PROGRAM

If you have an assembler, use the source code in **Listing 1** and save the object file as PRINTER.DUMP. If you don't have an assembler, type in the machine language from **Listing 2** and save it with

BSAVE PRINTER.DUMP,A$8000,L$2D8

You may need to modify the program slightly if your printer is attached to a slot other than slot one, or if your printer requires a line feed with every carriage return. To make the program use a different printer slot, type the following commands from the Monitor (where *n* is the desired slot number):

```
BLOAD PRINTER.DUMP
825C:Cn
BSAVE PRINTER.DUMP,A$8000,L$2D8
```

To alter the program to send line feeds with carriage returns, enter the following commands from the Monitor:

```
BLOAD PRINTER.DUMP
82A6:A9 0A 20 ED FD
BSAVE PRINTER.DUMP,A$8000,L$2D8
```

HOW IT WORKS

The program's operation can be broken into three main sections. The first section of the program executes only when you install the DUMP command by BRUNning PRINTER.DUMP. Its function is to relocate and install the code for the DUMP command between the buffers and the BASIC interpreter. It does this by first requesting a one page buffer with a call to GETBUFR ($BEF5). If an error is returned, the program prints an error message and exits. Otherwise, it saves the most significant byte of the buffer. It then saves the address of the next command, and puts it in XTERNCMD ($BE06-$BE07). Finally, it copies code to the buffer and exits to BASIC.

The second section gains control whenever ProDOS gets a command it does not recognize. Its function is to check the current command, which is pointed to by VPATH1 ($BE6C-$BE6D). If it is not the DUMP command, it sets the Carry flag and jumps to the next command routine. If it is DUMP, it tells the BASIC Interpreter that the command is valid by putting a zero in XCNUM ($BE53), the command's length minus one in XLEN ($BE52), and the address of our third section in XTERNADDR ($BE50-$BE51). Then it exits to the third section.

The third section does the actual printing of the screen. First it saves all of the variables that it uses and initializes the printer. Then it falls into two nested loops. The outer loop executes once for each line on the screen. The inner loop executes once for each pair of columns on the screen. The first column in each pair is in auxiliary memory for 80 columns. If the computer is in 40-column mode, this column is skipped. After executing the loops, the program restores the variables and exits.

Robert Heath, 725 East 35th St., Erie, PA 16500. *ProDOS is required for this command.*

LISTING 1: PRINTER.DUMP Source Code

```
1 * Printer Dump
2 * by Robert Heath
3 * Copyright(c) 1988
4 * MicroSPARC, Inc.
5 * Concord, MA  01742
6 *
7 *  This program will add an external command to
8 *  the ProDOS Basic Interpreter.  This program
9 *  is a 40/80 column screen dump.  By entering
10 *  'DUMP', whatever is on the screen will be
11 *  printed on your printer.
12 *
13 * Assembler: Apple ProDOS Editor-Assembler
14 *
15 *  Program Equates
16 *
17 *
18          ORG   $8000
19 CH       EQU   $24          :Horz. cursor position
20 CV       EQU   $25          :Vertical cursor position
21 BASL     EQU   $28          :Address of start of line
22 CSWL     EQU   $36          :Output hook
23 CSWH     EQU   $37
24 KSWL     EQU   $38          :Input hook
25 KSWH     EQU   $39
26 LINE     EQU   $3C          :Variable for line number
27 A1L      EQU   $3C          :Temp monitor variable
28 A1H      EQU   $3D
29 A2L      EQU   $3E
30 A2H      EQU   $3F
31 EXTERNCMD EQU  $BE07        :External command handler
32 VPATH1   EQU   $BE6C        :Pointer to command line
33 BIENTRY  EQU   $BE00        :Entry point to BI
34 XTERNADDR EQU  $BE50        :Start of external program
35 XLEN     EQU   $BE52        :Length of command
36 XCNUM    EQU   $BE53        :Number of command
37 PBITS    EQU   $BE54        :Parameters permitted
38 GETBUFR  EQU   $BEF5        :Allocate buffers
39 MACHID   EQU   $BF98        :Machine identification
40 KEYBOARD EQU   $C000        :Keyboard port
41 KBDSTROBE EQU  $C010        :Keyboard strobe
42 RD80VID  EQU   $C01F        :Test if in 80 column mode
43 *                           :   High bit set = 80 col
44 TXTPAGE1 EQU   $C054        :Access text page 1
45 TXTPAGE2 EQU   $C055        :Access text page 2
46 VTABZ    EQU   $FC24        :Set BASL to correct line
47 HOME     EQU   $FC58        :Clear the screen
48 COUT     EQU   $FDED        :Standard output routine
49 :
50 ***
51 *  Segment I
52 *  Initialization
53 *  This section installs the command
54 *  in the BI and copies the next two
55 *  sections to a free buffer.
56 ***
57 :
58          LDA   #$01         :Allocate 1 page
59          JSR   GETBUFR
60          BCS   ERROR
61          STA   A2H          :Save MSB for copy
62          STA   XADDRH+1     :Put MSB in section II
63          STA   COMPARE+4    :Save MSB in section II
64          LDA   EXTERNCMD    :Save the current pointer
65          STA   INVALID+2    : in section II to link the
66          LDA   EXTERNCMD+1  : external commands
67          STA   INVALID+3
68          LDA   A2H          :Install our command
69          STA   EXTERNCMD+1
70          LDA   #$00
71          STA   EXTERNCMD
72          STA   A2L          :Set LSB for copy
73          LDA   #<START      :Get the start of program
74          STA   A1H          :   and save for copy
75          LDA   #>START
76          STA   A1L
77          LDY   #$00
78 COPY     LDA   (A1L),Y      :Get a byte of the program
79          STA   (A2L),Y      : and put it in the buffer
80          INY
81          BNE   COPY         :Loop until we are done
82 QUIT.X   RTS
83 :
84 ***
85 *  Error Routine
86 ***
87 :
88 ERROR    LDX   #$00         :Print the error message
89 ERROR.1  LDA   TXT.ERROR,X
90          BEQ   ERROR.X
91          JSR   COUT
92          INX
93          BNE   ERROR.1
```

LISTING 1: PRINTER.DUMP Source Code

```
94 ERROR.X  PLA   :Cancel       return
95          PLA
96          JMP   BIENTRY      :Enter the BI
97 :
98          MSB   ON
99 TXT.ERROR DFB  $8D,$8D,$87
100         ASC   'NO          ROOM FOR THIS COMMAND.'
101         DFB   $8D,$8D,$00
102 :
103 :
104         DS    $8200-*      :Install padding
105 :
106 ***
107 *  Main Program
108 *  This segment is installed in the
109 *  buffer obtain by FREEBUFF.
110 *  Note:  This segment must be page aligned
111 ***
112 :
113         DO    >*
114         FAIL  1,Main       segment not page aligned.
115         FIN
116 :
117 ***
118 *  Segment II
119 *  Command Handler
120 *  This section determines if 'DUMP'
121 *  was entered.  If it was, then it
122 *  tells the BI to execute segment III.
123 *  Otherwise, it jumps to the next
124 *  command handler.
125 ***
126 :
127 START    CLD                :Identify this as a command
128         LDA   VPATH1       :Point to the command
129         STA   A1L
130         LDA   VPATH1+1
131         STA   A1H
132         LDY   #$04
133 COMPARE  LDA   (A1L),Y
134         CMP   TXT.DUMP-1,Y
135         BNE   INVALID
136         DEY
137         BNE   COMPARE
138         LDA   #$00         :No parameters needed
139         STA   PBITS
140         STA   PBITS+1
141         STA   XCNUM        :This is an external command
142         LDA   #$03         :Length of command - 1
143         STA   XLEN
144         LDA   #>MAIN       :Point to segment III
145         STA   XTERNADDR
146 XADDRH  LDA   #$00         :00 changed by segment I
147         STA   XTERNADDR+1
148         CLC
149         RTS
150 INVALID  SEC                :It's not us, pass to the
151         JMP   $0000        :   next command handler
152 :
153 ***
154 *  Segment III
155 *  Dump Routine
156 *  This section does the printing
157 *  of the screen
158 ***
159 :
160 MAIN     EQU   *
161         LDA   CSWL         :Save the important values
162         PHA
163         LDA   CSWH
164         PHA
165         LDA   KSWL
166         PHA
167         LDA   KSWH
168         PHA
169         LDA   CH
170         PHA
171         LDA   CV
172         PHA
173         LDX   #$FF         :Initialize 80 col flag
174         LDA   MACHID       :Check to see if a 80 col
175         AND   #$02         :   card is installed
176         BEQ   MAIN40       :No,must be in 40 col mode
177         BIT   RD80VID      :In 80 col mode?
178         BPL   MAIN40       :No, in 40 col mode
179         LDX   #$00         :Yes,set flag for 80 col
180 MAIN40   EQU   *
181         TXA                :Transfer flag to stack
182         PHA                : 0=80 Col   $FF=40 Col
183         FIN
184         LDA   #PRINTER     :Set output hook 2 printer
185         STA   CSWH
186         LDA   #$00
187         STA   CSWL
188         TAX                :Print the init. string
189 INIT     LDA   STRING,X     :Get a character
190         BEQ   INITX        :Exit if zero
191         JSR   COUT         :Print the character
192         INX
193         BNE   INIT         :Always branch
194 INITX    LDA   #$FF
195         STA   LINE
196 PRLINE   INC   LINE         :Move to the next line
197         LDA   LINE
198         STA   CV
199         JSR   VTABZ        :Set BASL to point to line
200         LDY   #$00         :Initialize to column 1
201 LOOP     LDA   KEYBOARD     :See if ESC was pressed
202         CMP   #$9B
203         BEQ   EXIT         :Yes, stop printing here
204         PLA                :In 40 col mode?
205         PHA
```

```
206             BNE   COL40           ;Yes, forget page 2
207             PHP                   ;Save interrupt status
208             SEI                   ;Turn off the interrupts
209             STA   TXTPAGE2        ;Access text page 2
210             LDA   (BASL),Y        ;Get a char from page 2
211             STA   TXTPAGE1        ;Access text page 1
212             PLP                   ;Restore interrupt status
213             JSR   COUT            ;Print the character
214 COL40       LDA   (BASL),Y        ;Get a char from page 1
215             JSR   COUT            ;  and print it.
216             INY                   ;Go to the next column
217             CPY   #$28            ;End of line?
218             BNE   LOOP            ;No, continue with line
219             LDA   #$8D            ;Yes, Print a RETURN
220             JSR   COUT
221 *           LDA   #$0A
222 *           JSR   COUT
223             NOP                   ;Leave room for code that
224             NOP                   ;  can print a line feed
225             NOP
226             NOP
227             NOP
228             LDA   LINE
229             CMP   #$17            ;End of the screen?
230             BNE   PRLINE          ;No, do the next line
231 EXIT        PLA                   ;Clear flag from stack
232             PLA
233             STA   CV
234             JSR   VTABZ
235             PLA
236             STA   CH
237             PLA
238             STA   KSWH
239             PLA
240             STA   KSWL
241             PLA
242             STA   CSWH
243             PLA
244             STA   CSWL
245             STA   KBDSTROBE       ;Clear the strobe
246             RTS                   ;We're done,let's exit
247 ;
248 ***
249 *  PROGRAM VARIABLES
250 ***
251 ;
252             MSB   ON
253 STRING      DFB   $8D,$8D,$89     ;Set line size to 81 col.
254             ASC   '82N'           ;  and print blank lines
255             DFB   $8D,$8D,$00     ;80 columns + C.R. + L.F.
256 PRINTER     EQU   $C1             ;MSB of printer addr.; use $Cx if
257             MSB   OFF             ;  your printer is in slot x
258 TXT.DUMP    ASC   'DUMP'
```

END OF LISTING 1

LISTING 2: PRINTER.DUMP

```
Start: 8000              Length:2D8

D7 8000:A9 01 20 F5 BE B0 32 85
24 8008:3F 8D 2D 82 8D 11 82 AD
58 8010:07 BE 8D 35 82 AD 08 BE
A8 8018:8D 36 82 A5 3F 8D 08 BE
D5 8020:A9 00 8D 07 BE 85 3E A9
CD 8028:82 85 3D A9 00 85 3C A0
4C 8030:00 B1 3C 91 3E C8 D0 F9
6F 8038:60 A2 00 BD 4B 80 F0 06
69 8040:20 ED FD E8 D0 F5 68 68
1D 8048:4C 00 BE 8D 8D 87 CE CF
9B 8050:A0 D2 CF CF CD A0 C6 CF
74 8058:D2 A0 D4 C8 C9 D3 A0 C3
ED 8060:CF CD CD C1 CE C4 AE 8D
82 8068:8D 00 C5 A0 A0 A0 A0 F2
C0 8070:A0 A0 E3 A0 A0 A0 B7 A0
CE 8078:C5 E7 A0 CA EF A0 A0 A0
74 8080:B6 A0 A0 B0 D3 C3 A0 A0
46 8088:A0 A0 A0 A0 A0 A0 A0 A0
9E 8090:A0 A0 A0 B1 A0 A0 A0 A0
97 8098:A0 A0 A0 A0 B5 F2 A0 A0
51 80A0:A0 A0 A0 D4 A0 A0 CC C2
6E 80A8:A0 A0 A0 A0 A0 A0 A0 A0
C7 80B0:B8 A0 A0 C5 A0 CF A0 A0
D0 80B8:C4 A0 A0 A0 EC A0 A0 A0
95 80C0:A0 A0 A0 A0 A0 D0 A0 A0
46 80C8:A0 B3 A0 BB A0 A0 A0 FF
55 80D0:A0 A0 81 C4 A0 A0 C4 B3
42 80D8:C1 A0 A0 C8 A0 A0 A0 A0
92 80E0:80 A0 A0 80 A0 A0 A0 A0
97 80E8:A0 A0 A0 A0 A0 A0 B9 A0
EF 80F0:B0 FF A0 CC B6 A0 D0 FF
C2 80F8:A0 D2 80 A0 C9 A0 A0 A0
8D 8100:A0 B0 F4 C5 A0 A0 A0 A0
44 8108:A0 C5 A0 F2 A0 A0 A0 A0
28 8110:B0 D9 A0 CC A0 A0 A0 A0
D4 8118:A0 A0 B8 A0 A0 A0 A0 A0
D8 8120:B8 C3 A4 A0 C9 C3 A0 C4
FD 8128:A0 A0 A0 B1 A0 A0 A0 A0
20 8130:A0 C1 A0 A0 A0 A0 A0 A0
D6 8138:A0 A0 A0 A0 B0 A0 C3 80
B2 8140:A0 A0 D2 C2 D1 A0 A0 A0
6C 8148:A0 A0 A0 A0 A0 FF A0 A0
ED 8150:8D C4 EF A0 A0 B6 A0 A0
1D 8158:A0 A0 A0 A0 A0 A0 80 A0
5B 8160:D8 80 A0 D2 A0 A0 A0 80
E2 8168:EF A0 A0 F7 A0 A0 A0 A0
A9 8170:A0 A0 B9 A0 A0 FF A0 A0
70 8178:80 A0 A0 A0 A0 A0 A0 A0
10 8180:A0 A0 A0 A0 A0 A0 BB A0
A6 8188:B0 D9 A0 CC A0 A0 A0 A0
22 8190:A0 A0 B8 A0 A0 A0 A0 A0
85 8198:C3 A0 A0 A0 C9 C2 A0 A0
B7 81A0:A0 A0 A0 B1 BB A0 A0 A0
DE 81A8:A0 C1 A0 A0 A0 A0 A0 A0
F9 81B0:A0 B9 A0 A0 B0 A0 A0 80
5E 81B8:A0 A0 80 C2 BB A0 A0 A0
EE 81C0:D3 A0 A0 A0 A0 FF B0 EE
95 81C8:8D A0 A0 A0 A0 B6 A0 A0
FE 81D0:A0 A0 A0 C1 A0 A0 80 A0
6D 81D8:D8 80 A0 A0 80 A0 C5 80
54 81E0:A0 AE A0 A0 A0 A0 A0 A0
67 81E8:A0 A0 B9 D2 A0 FF A0 EF
BE 81F0:80 A0 A0 A0 A0 A0 A0 A0
6A 81F8:A0 A0 A0 B0 A0 A3 80 A0
B7 8200:D8 AD 6C BE 85 3C AD 6D
67 8208:BE 85 3D A0 04 B1 3C D9
0D 8210:D3 82 D0 1F 88 D0 F6 A9
47 8218:00 8D 54 BE 8D 55 BE 8D
76 8220:53 BE A9 03 8D 52 BE A9
B5 8228:37 8D 50 BE A9 00 8D 51
63 8230:BE 18 60 38 4C 00 00 A5
13 8238:36 48 A5 37 48 A5 38 48
4F 8240:A5 39 48 A5 24 48 A5 25
FE 8248:48 A2 FF AD 98 BF 29 02
AC 8250:F0 07 2C 1F C0 10 02 A2
DD 8258:00 8A 48 A9 C1 85 37 A9
49 8260:00 85 36 AA BD CB 82 F0
30 8268:06 20 ED FD E8 D0 F5 A9
DE 8270:FF 85 3C E6 3C A5 3C 85
5C 8278:25 20 24 FC A0 00 AD 00
09 8280:C0 C9 9B F0 2C 68 48 D0
1C 8288:0E 08 78 8D 55 C0 B1 28
21 8290:8D 54 C0 28 20 ED FD B1
87 8298:28 20 ED FD C8 C0 28 D0
FB 82A0:DD A9 8D 20 ED FD EA EA
EC 82A8:EA EA EA A5 3C C9 17 D0
4A 82B0:C2 68 68 85 25 20 24 FC
2F 82B8:68 85 24 68 85 39 68 85
07 82C0:38 68 85 37 68 85 36 8D
42 82C8:10 C0 60 8D 8D 89 B8 B2
88 82D0:CE 8D 8D 00 44 55 4D 50

TOTAL: 24F8
```

END OF LISTING 2